

COMMODORE

MENSILE - SPED. AB. POST. GR. III/70 - N.° 2 - GENNAIO 1985 - EDIZIONI CO.STA - PISA



dossier COMMODORE

Mensile

di giochi e programmi
per COMMODORE 64
e VIC 20

Direttore Responsabile:

Marco Bindi

Direttore Tecnico:

Gianroberto Negri

Edizioni **CO.STA** srl

via Mazzini 128

56100 PISA

Registrazione al Tribunale
di Pisa n. 2 del 22/2/84
Spediz. in abbon. post.
gruppo III / 70

COMMODORE DOSSIER

il mensile per realizzare la tua
Biblioteca di Software in tre
sezioni:

- **Giochi**
- **Programmi personali**
- **Programmi di utilità**

Lire 2000

I numeri arretrati, a lire 4000,
possono essere richiesti sen-
za oneri di spedizione con pa-
gamento anticipato in contan-
ti, o tramite assegno o vaglia
postale intestato alla Società
editrice.

Impianti:

Cencograf

Stampa:

Rotografica srl (MI)

Distribuzione:

Messaggerie Periodici (MI)

dossier

**Ogni mese in edicola
il DOSSIER
per il tuo computer**

- **COMMODORE DOSSIER**
- **SINCLAIR DOSSIER**
- **TEXAS 99/4A DOSSIER**

**giochi, programmi di utilità
e programmi personali per
realizzare una fantastica
biblioteca software**



UTILITY



**PROGRAMMI
PERSONALI**



GIOCHI

CODIFICA SIMBOLI

COMMODORE



PIANTA



Avete mai provato a cimentarvi nella coltivazione delle piante?... Non si tratta di una impresa semplice, lo testimonia il fatto che riuscire a mantenere in vita la pianta simulata da questo programma. Mediante esso, potrete infatti mettere alla prova le Vostre capacita' di floricultori. Al lancio del programma, viene richiesto in quale zona desiderate porla, se a Sud, a Nord oppure ad Est. Per rispondere, bastera' premere la lettera iniziale della parola (Es. N per Nord); fatto questo Vi si presentera' un menu' composto da alcune opzioni, precedute da delle lettere in REVERSE. E' premendo tali lettere che sara', possibile effettuare la scelta delle opzioni. A questo punto stara' a Voi decidere, quando annaffiarla (BAGNA), quando concimarla (CONCIMA), quando cambiarla di posto (LOCAZIONE), quando metterla in un vaso piu' grande (VASO) o quando lasciarla semplicemente crescere (NIENTE). Buon lavoro...

90 REM

```
100 IS=10:DW=0:DF=0:DL=0:SIZE=0:CW=0:CF=0:CL=0:CS=0:POKE53281,1:D
AY=0:LEAF=1
```

```
105 LW=0:LF=0:LS=0:BL=INT(RND(1)*25+25):POT=1:CH=55296:SC=1024:R$
=CHR$(13)
```

```
110 PRINT"[CLR][BLU]DOVE VUOI METTERE LA PIANTA (FIN.)":INPUT"[RV
S ON][RVS OFF]ORD ,[RVS ON][RVS OFF]UD[2 SPZ],[RVS ON][RVS OFF]
1ST ";LOC$
```

```
120 IF LOC$="N"THENIW=14:IP=14:IL=7:POKE53281,6:GOTO160
```

```
130 IF LOC$="S"THENIW=3:IP=7:IL=7:POKE53281,7:GOTO160
```

```
140 IF LOC$="E"THENIW=7:IP=10:IL=30:POKE53281,14:GOTO160
```

```
150 GOTO110
```

```
160 DAY=DAY+1
```

COMMODORE

```

170 PRINT"[CLR][NERO]GIO."DAY,R$"[GIU']COMANDO"R$"[2 GIU'][RVS ON
J][RVS OFF]BAGNA"R$"[RVS ON][RVS OFF]CONCIMA"R$"[RVS ON][RVS OF
F]OCCAZIONE"
180 PRINT"[RVS ON][RVS OFF]VASO"R$"[RVS ON][RVS OFF]NIENTE[2 GI
U']":INPUTC$
190 ON-(C$="W")-2*(C$="F")-3*(C$="L")-4*(C$="R")GOTO200,300,400,5
00
191 IF C$="D" THEN GOTO 1000
195 GOTO170
200 LW=0:IF DW>=IWHENDW=0:GOTO1000
220 PRINT"ANNAFFIATA ":DW=0:CW=CW+1:IFCW>3THEND$="ATTECHITO":GOTO
3000
230 GOTO2000
300 LF=0:IFDF>=IPTHENDF=0:GOTO1000
320 PRINT"CONCIMATA":DF=0:CF=CF+1:IFCF>3THEND$="ESSICATA":GOTO300
0
330 GOTO2000
400 IF LOC$="N"THENPRINT"[RVS ON][RVS OFF]UD O [RVS ON][RVS OFF
]ST (FINESTRA"
420 IF LOC$="S"THENPRINT"[RVS ON][RVS OFF]ST O [RVS ON][RVS OFF
]ORD (FINESTRA"
430 IF LOC$="E"THENPRINT"[RVS ON][RVS OFF]UD O [RVS ON][RVS OFF
]ORD (FINESTRA"
440 INPUTLOC$:IFLOC$="N"ORLOC$="S"ORLOC$="E"THEN450
445 GOTO400
450 CL=CL+1:IFCL>5THEND$="ABBOZZO DI MOSSA[2 SPZ]":GOTO3000
460 GOTO120
500 LS=0:IFSIZE>=ISTHENIS=IS+10:POT=POT+1:GOTO1000
520 PRINT"TROPPO GROSSO":POT=POT+1:CS=CS+1:IFCS>3THEND$="ESAGERAT
O!!":GOTO3000
530 GOTO2000
600 IFPOT>0THENPOKECH+826,4:POKESC+826,160:POKECH+866,4:POKESC+86
6,160
615 IFPOT>0THENPOKECH+825,4:POKESC+825,95:POKECH+827,4:POKESC+827
,105
620 IFPOT>1THENPOKESC+825,160:POKESC+827,160:POKESC+865,160:POKEC
H+865,4
625 IFPOT>1THENPOKESC+867,160:POKECH+867,4:POKESC+824,95:POKECH+8
24,4
626 IFPOT>1THENPOKESC+828,105:POKECH+828,4
630 IFPOT>2THENPOKECH+824,4:POKESC+824,160:POKECH+828,4:POKESC+82
8,160

```

COMMODORE

```

635 IFPOT>2THENPOKECH+864,4:POKESC+864,160:POKECH+868,4:POKESC+86
8,160
640 IFPOT>3THENPOKECH+823,4:POKESC+823,160:POKECH+829,4:POKESC+82
9,160
650 FORI=1TOLEAF:READP:POKESC+P,65:POKECH+P,5:NEXT:RESTORE
670 PRINT"[4 SPZ][GIU'] [RVS ON]UN TASTO -->[RVS OFF]"
675 IFPOT>3THENPRINT"VASO LARGO"
680 GETK$:IFK$=""THEN680
690 RETURN
700 IFLOC$="S"THENLEAF=LEAF+INT(RND(1)*3+1)
720 IFLOC$="N"THENLEAF=LEAF+INT(RND(1)*2)
730 IFLOC$="E"THENLEAF=LEAF+INT(RND(1)*2+1)
740 RETURN
1000 IF DW<IW AND DF<IPAND DL<IL AND SIZE<IS THEN 1030
1020 GOTO1040
1030 GOSUB700:DW=DW+1:DF=DF+1:DL=DL+1:SIZE=SIZE+1:PRINT"STA' CRES
CENDO"
1031 IFLEAF>BLTHEN2500
1032 GOSUB600:GOTO160
1040 IFDW>=IWTHENPRINT"TROPPO ARIDA ":LW=LW+1:IFLW>3THEND$="HA SE
TE":GOTO3000
1050 IFDF>=IPTHENPRINT"VUOLE CONCIME!":LF=LF+1:IFLF>3THEND$="HA F
AME":GOTO3000
1060 IFSIZE>=ISTHENPRINT"TROPPO P.":LS=LS+1:IFLS>3THEND$="SOFFOCA
":GOTO3000
1080 IFLOC$="N"ANDDL>=ILTHENPRINT"NON CI SIAMO":D$="TROPPO PICCOL
A":GOTO1100
1085 IFLOC$="E"ANDDL>=ILTHENPRINT"E' INFELICE":D$="POCO AMORE":GO
TO1100
1090 IFLOC$="S"ANDDL>=ILTHENPRINT"STA' ESSICANDO":D$="ESAGERATO":
GOTO1100
1095 GOTO1110
1100 J=RND(1)*10:IFJ<2THEN3000
1105 GOTO2000
1110 GOSUB600:GOTO160
1200 DATA786,746,785,787,706,744,748,745,747,666,705,707,703,709,
704,708,784
1210 DATA788,743,749,665,667,626,783,789,742,750,702,710,664,668,
625,627,663
1220 DATA669,586,624,628,782,790,701,711,585,587,546,662,670,623,
629,584,588
1230 DATA506,545,547
2000 GOSUB600:LEAF=LEAF-1:IFLEAF<1THEND$="PECCATO .....":GOTO300

```

COMMODORE

```

0
2020 GOTO160
2500 PRINT"[CLR][GIU' ][8 SPZ]CONGRATULAZIONI!"R$"[3 GIU' ]LA VOSTR
A PIANTA HA"DAY"GIORNI"
2520 FORI=1TO3000:NEXT
2530 PRINT"[CLR][GIU' ][10 SPZ][VRD][SHFT A]"
2540 PRINT"[9 SPZ][SHFT A][SHFT B][SHFT A]"R$"[8 SPZ][SHFT A][ROS
S][SHFT Q][VRD][SHFT A][SHFT N][SHFT A]"R$"[7 SPZ][SHFT A][SHFT M
][SHFT A][SHFT B][SHFT A][ROSS][SHFT Q][VRD][SHFT A]"
2550 PRINT"[6 SPZ][SHFT A][ROSS][SHFT Q][VRD][SHFT A][SHFT M][SHF
T A][SHFT N][SHFT A][SHFT N]"R$"[5 SPZ][SHFT A][SHFT M][SHFT A][S
HFT M][SHFT A][SHFT B][SHFT A][ROSS][SHFT Q][VRD][SHFT A][SHFT N]
[SHFT A]"R$"[4 SPZ][SHFT A][SHFT M][SHFT A][ROSS][SHFT Q][VRD][SH
FT A][SHFT M][SHFT A][SHFT N][SHFT A][SHFT N][SHFT A][SHFT N][SHF
T A]"
2560 PRINT"[6 SPZ][SHFT A][SHFT M][SHFT A][SHFT M][SHFT A][ROSS][
SHFT Q][VRD][SHFT A][SHFT N]"R$"[5 SPZ][SHFT A][SHFT M][SHFT A][S
HFT M][SHFT A][ROSS][SHFT Q][VRD][SHFT A][SHFT N][SHFT A][SHFT N]
[SHFT A]"R$"[4 SPZ][SHFT A][ROSS][SHFT Q][VRD][SHFT A][SHFT M][SH
FT A][SHFT M][SHFT A][SHFT N][SHFT A][SHFT N][SHFT A][ROSS][SHFT
Q][VRD][SHFT A]"
2570 PRINT"[6 SPZ][SHFT A][SHFT M][SHFT A][ROSS][SHFT Q][VRD][SHF
T A][SHFT N][SHFT A][SHFT N]"R$"[5 SPZ][SHFT A][SHFT M][SHFT A][S
HFT M][SHFT A][SHFT B][SHFT A][ROSS][SHFT Q][VRD][SHFT A][SHFT N]
[SHFT A]"R$"[4 SPZ][SHFT A][ROSS][SHFT Q][VRD][SHFT A][SHFT M][SH
FT A][SHFT M][SHFT A][SHFT N][SHFT A][SHFT N][SHFT A][ROSS][SHFT
Q][VRD][SHFT A]"
2580 PRINT"[8 SPZ][BLU][CMD *][RVS ON][3 SPZ][RVS OFF][SHFT ST.]
"R$"[9 SPZ][RVS ON][3 SPZ][RVS OFF][NERO]"
2720 PRINT"[GIU' ][3 SPZ][RVS ON]PER GIOCARE[2 SPZ][RVS OFF]":GOSU
B670:GOTO100
3000 PRINT"E' MORTA .....":D$
3020 PRINT"[GIU' ][3 SPZ][RVS ON]PER GIOCARE[2 SPZ][RVS OFF]":GOSU
B600:GOTO100

```



TEST-VITA



Vi siete mai chiesti se il tipo di vita che conducet

COMMODORE

influisca sulla sua durata?... Rispondendo accuratamente alle domande fatteVi dal computer, in base a calcoli di natura statistica, potrete avere un indicazione di quanto potrete ancora vivere. Ovviamente il tutto e' estremamente indicativo e non va' preso come oro colato. Questo a causa anche del fatto che, secondo la statistica se io sto mangiando un pollo ed un mio amico mi sta guardando, abbiamo mangiato mezzo pollo a testa!!... ASSURDO...

```

1 REM VITA *****
2 PRINTCHR$(14)
3 L=100
5 PRINT"[CLR]"
10 PRINT"[NERO][SHFT N]OME PREGO":INPUTN$
20 PRINT"[SHFT E]TA' PREGO":INPUTAGE
25 PRINT"[CLR]";PRINT"";PRINT"[SHFT S]EI MASCHIO O FEMMINA":PRINT
"";PRINT"([SHFT M] O [SHFT F])
27 INPUTSE$
28 IFSE$="M"THENLETSE=69
29 IFSE$="F"THENLETSE=74
30 BIRTH=1985-AGE
100 PRINT" [RVS ON] COMPLEANNO,[RVS OFF]"N$;" "
110 PRINT"NATO NEL "BIRTH:FORI=1TO2000:NEXTI
200 PRINT"[CLR]";"[2 SPZ][SHFT S]EI SPOSATO";INPUT Y$
210 IFY$="S"ORY$="SI"THENLETL=L-1:GOTO220
212 IFY$="N"ORY$="NO"THEN220
215 IFY$<>"S"THENPRINT"[2 SPZ]PREGASI RIPETERE ":FORU=1TO2000:NEXT:GOTO200
220 PRINT"[CLR]";"";PRINT"[SHFT S]EI RICCO [SHFT P]OVERO O VAI ME
DIAMANTE(R-P-M)"
230 INPUTR$
240 IFR$="R"ORR$="P"THENLETL=L-2:GOTO250
242 IFR$="M"THENLETL=L-1:GOTO250
243 IFR$<>"P"THENPRINT" [SHFT R]ISPOSTA NON VALIDA":FORG=1TO2000:
NEXT:GOTO220
250 PRINT"[CLR] [SHFT S]EI GRASSO[8 SPZ]";PRINT:PRINT"([SHFT S] O
R [SHFT N])"
260 INPUTW$
270 IFW$="S"ORW$="SI"THENLETL=L-1:GOTO280
272 IFW$<>"N"THENPRINT" [SHFT R]ISPOSTA NON VAL.":FORAA=1TO2000:N

```

COMMODORE

```

EXT:GOTO250
280 PRINT"[CLR]"
290 PRINT"[SHFT FJAI ESERCIZIO[8 SPZ]]":PRINT"":PRINT"(1-[SHFT SJ
EMPRE 2-[SHFT MJAI[7 SPZ]]-[SHFT AJ VOLTE[2 SPZ]]"
300 INPUTEX
310 IFEX=2THENLETL=L-1:GOTO330
320 IFEX=3THENLETL=L-0:GOTO330
322 IFEX<>1THENPRINT"[SHFT NJON VALIDA[6 SPZ]]":FORV=1TO2000:NEXT
:GOTO280
330 PRINT"[CLR]":PRINT"[SHFT BJEVI[8 SPZ]]":PRINT"":PRINT"(1-[SHFT
MJODERATAMENTE[6 SPZ]]2-[SHFT MJOLTO[2 SPZ]]":PRINT""
340 PRINT" 3-[SHFT MJAI[2 SPZ]]"
350 INPUTD
360 IFD=2THENLETL=L-1:GOTO400
370 IFD=1THENLETL=L-0:GOTO400
380 IFD<>3THENPRINT"[SHFT NJON VALIDA[6 SPZ]]":FORVV=1TO2000:NEXT
:GOTO330
400 PRINT"[CLR]":PRINT"[SHFT FJUMI[8 SPZ]]([SHFT SJ OR [SHFT NJ))"
410 INPUTSM$
420 IFSM$="S"ORSM$="SI"THENLETL=L-1:GOTO422
421 IFSM$<>"N"THENPRINT"[SHFT NJON VALIDA[6 SPZ]]":FORSQ=1TO2000:
NEXT:GOTO400
422 IF SM$="S" OR SM$="SI" THEN PRINT:PRINT"[SHFT MJOLTE AL GIORN
O ";: INPUT HW
423 IFHW>20THENLETL=L-2
424 IFHW<20THENLETL=L-1
425 IFHW>50THENLETL=L-2
430 PRINT"[CLR]":PRINT"[SHFT TJI AMMALI SPESSO ":PRINT"":PRINT"[
2 SPZ]]([SHFT SJ OR [SHFT NJ))"
440 INPUTIL$
450 IFIL$="S"ORIL$="SI"THENLETL=L-1:GOTO455
451 IFIL$<>"N"THENPRINT"[SHFT NJON VALIDA[6 SPZ]]":FORXX=1TO2000:
NEXT:GOTO430
455 PRINT"[CLR]"
460 PRINT:PRINT"[SHFT SJEI SPESSO TESO[4 SPZ]]":PRINT"":PRINT"(S
O N)"
470 INPUTT$
480 IFT$="S"ORT$="SI"THENLETL=L-1:GOTO498
485 IFT$<>"N"THENPRINT"[SHFT NJON VALIDA[6 SPZ]]":FORSAX=1TO2000:
NEXT:GOTO455
498 VOL=36878
499 POKE36878,15
500 PRINT"[CLR]":FORVV=128TO140:POKE36879,VV:POKEVOL,15:POKE36875

```

COMMODORE

```
,210:FORG=1T0100:NEXTG
503 POKE36878,0:NEXTVV
510 POKE36879,122
520 PRINT"[NERO]****[RVS ON]VITA RIMASTA[3 SPZ][RVS OFF]****"
530 PRINT:PRINT
540 XX=(L/100)*SE
550 X1=XX
560 PRINT"[2 SPZ][SHFT D]OVRESTI POTER VIVERE FINO A "X1
570 X3=INT(XX-AGE)
580 IFX3<0THENPRINT"[2 SPZ][SHFT T]I SEI BRUCIATO GLI[5 SPZ]ANNI
RIMASTI[3 SPZ]---":END
590 PRINT"*****[23 COMD I]"
600 PRINT:PRINT:PRINT"[NERO][COMD G]HAI ANCORA "X3[4 SPZ]"[2 SPZ]
DA VIVERE!!... "
```



SILLABARIO



Questo programma Vi permettera' di insegnare la corretta definizione delle parole, ad esempio alla definizione : "corre dietro ad i gatti " dovra' essere risposto ovviamente CANE. Non solo ma esso Vi permettera' d'insegnare come si scrive correttamente. In caso d'errore da parte del bimbo, il computer provvedera' a segnalarlo. Sempre in caso d'errore ,premendo il tasto F1 potra' essere visualizzata la prossima lettera della risposta corretta. A titolo d'esempio sono riportate nei DATA in fondo al programma alcune parole con le rispettive definizioni; sara' compito Vostro cercarne delle altre ed aggiungerle in coda ad esse.

```
5 REM SILLABARIO *****
10 POKE53281,1:POKE53280,1:PRINTCHR$(144);CHR$(14);CHR$(147)
15 CLR
20 S=54272:V=53248
```

COMMODORE

```

25 RESTORE
49 REM **** TITOLO *****
50 FOR X= 1 TO 6:PRINT:NEXT X
55 PRINTTAB( 7)"[SHFT I]L SILLABARIO PER TUTTI"
60 PRINT:PRINTTAB(13)"*****"
70 PRINT:PRINT:PRINT:PRINT
75 PRINTCHR$(18)"[3 SPZ][3 SHFT SPZ][SHFT N][SHFT O][SHFT T][SHFT
E][5 SHFT SPZ]"
80 PRINT:PRINTTAB(4)" [SHFT C][SHFT A][SHFT M][SHFT B][SHFT I][SHF
T A][SHFT M][SHFT E][SHFT N][SHFT T][SHFT O][SHFT SPZ][SHFT P][SHF
T A][SHFT R][SHFT O][SHFT L][SHFT E][SHFT SPZ]- [SHFT F]1 "
85 FORX=1TO40:PRINTCHR$(183);:NEXTX:GOSUB2000
87 FORPR=0TO62:READPQ:POKE832+PR,PQ:NEXTPR
90 FORX=1TO100
92 GETF1$:IFF1$=""THEN95
93 IFF1$=CHR$(133)THEN2500
95 NEXTX
99 REM *** VISUALIZZAZIONE PAROLE ***
100 PRINTCHR$(147)
105 PRINT:PRINTCHR$(18)" PAROLE DA SILLABARE ":PRINTCHR$(146)
110 READA1$
120 IFA1$="-1"THEN150
125 READB1$,A2$,B2$
130 PRINTA1$:GOSUB2000
135 PRINTCHR$(145);TAB(20)A2$:GOSUB2000
140 GOTO110
150 PRINT:PRINTCHR$(18)" [SHFT P][SHFT R][SHFT E][SHFT M][SHFT E][
SHFT R][SHFT E] UN TASTO[16 SPZ]"
155 PRINT:PRINT:PRINTTAB(3)"[SHFT C][SHFT A][SHFT M][SHFT B][SHFT
I][SHFT A][SHFT M][SHFT E][SHFT N][SHFT T][SHFT O][SHFT SPZ][SHFT
D][SHFT E][SHFT F][SHFT I][SHFT N][SHFT I][SHFT Z][SHFT I][SHFT O]
[SHFT N][SHFT E] [SHFT F]1"
158 FORX=1TO40:PRINTCHR$(183);:NEXTX:GOSUB2000
160 GETF1$:IFF1$=""THEN160
165 IFF1$=CHR$(133)THEN2500
190 RESTORE:FORPR=0TO62:READPQ:NEXTPR
200 PRINTCHR$(147)
210 READA$,B$:CC=0
215 IFA$="-1"GOTO1500
219 REM *** VISUALIZZAZIONE DI B$ 220-245
225 FORX=1TO6:PRINT:NEXTX
227 PRINT:PRINT
229 PRINT"PAROLE DA SILLABARE"

```

COMMODORE

```

230 FORX=1TO40:PRINTCHR$(183);:NEXTX:PRINT:PRINT
232 FORI=0TOLEN(B$)
235 POKES+24,6:POKES+1,126:POKES,6:POKES+5,120:POKES+6,30:POKES+4
33
240 PRINTLEFT$(B$,I+1):POKES+24,0
245 PRINTCHR$(145);CHR$(145):FORX=1TO40:NEXTX
247 POKES+1,0:POKES,0:POKES+5,0:POKES+6,0:POKES+4,0:NEXTI
270 PRINT:PRINT:FORX=1TO100:NEXTX
280 INPUT"SILLABAZIONE PAROLA";C$
290 FORX=1TO50:NEXTX
300 IFC$=A$GOTO600
310 IFC$<>A$GOTO800
599 REM *** ROUTINE DI VINCITA ***
600 PRINTCHR$(147):FORX=1TO9:PRINT:NEXTX
605 SM=24
610 POKEV+4,200:POKEV+21,4:POKE2042,13:POKEV+29,4
612 POKEV+41,0
620 PRINT;A$:FORX=1TO200:NEXTX
622 PRINTCHR$(152);:FORX=1TO7:PRINTCHR$(178);:NEXTX
623 PRINT"SEI SULLA BUONA STRADA";:FORX=1TO6:PRINTCHR$(178);:NEXTX
624 PRINTCHR$(144)
625 FORTN=200TO24STEP-8
644 POKES+24,15:POKES+1,145:POKES,0:POKES+5,0:POKES+6,240:POKES+4,
129
645 POKEV+4,TN:POKEV+5,120
646 IFSM<1THENSM=1
647 SM=SM-1
648 GOSUB750
650 FORPU=1TO40:NEXTPU
651 POKES+24,0:POKES+1,0:POKES,0:POKES+5,0:POKES+6,0:POKES+4,0:NEX
TTN
653 POKES+24,15:POKES+1,155:POKES,255:POKES+5,0:POKES+6,240:POKES+
4,17
655 PRINTCHR$(19):FORX=1TO6:PRINT:NEXTX:PRINTTAB(14)"*****"
657 FORX=1TO500:NEXTX
658 POKES+24,0:POKES+1,0:POKES,0:POKES+5,0:POKES+6,0:POKES+4,0
659 FORX=1TO300:NEXTX
660 POKES+24,15:POKES+1,155:POKES,255:POKES+5,0:POKES+6,240:POKES+
4,17
661 FORX=1TO500:NEXTX
662 POKEV+21,0
663 POKES+24,0:POKES+1,0:POKES,0:POKES+5,0:POKES+6,0:POKES+4,0
664 PRINTCHR$(147):FORX=1TO6:PRINT:NEXTX

```

COMMODORE

```

670 PRINTCHR$(18)A$;CHR$(146);"QUESTA E' LA CORRETTA SILLABAZIONE"
675 GOSUB2000
680 FORX=1TO1000:NEXTX:PRINTCHR$(147):FORX=1TO6:PRINT:NEXT
700 GOTO200
730 REM **** ROUTINE PER IL FUMO ****
750 PRINTCHR$(19);:FORX=1TO8:PRINT:NEXTX
760 PRINTTAB(5M);CHR$(31);CHR$(42);CHR$(32);CHR$(42)
780 PRINTCHR$(144)
790 RETURN
799 REM
800 PRINTCHR$(147)
801 FORX=1TO6:PRINT:NEXTX
802 PRINT"[SHFT L]A VOSTRA RISPOSTA E'";CHR$(18);C$;CHR$(146);" ER
RATA"
805 REM
810 POKES+24,15:POKES+1,3:POKES,255:POKES+5,0:POKES+6,240:POKES+4,
33
820 FORX=1TO600:NEXTX
825 POKES+24,15:POKES+1,34:POKES,75:POKES+5,128:POKES+6,128:POKES+
4,39
830 FORX=1TO600:NEXTX
840 POKES+24,0:POKES+1,0:POKES,0:POKES+5,0:POKES+6,0:POKES+4,0
865 PRINT:PRINT"MOLTO MALE"
870 FORX=1TO800:NEXTX
880 PRINTCHR$(147):FORX=1TO6:PRINT:NEXTX
890 PRINTB$:PRINT:FORX=1TO500:NEXTX
892 PRINTCHR$(18);"[SHFT P]ER LA PROSSIMA LETTERA PREMERE [SHFT F]
1 "
893 GETCL$:IFCL$=""THEN893
894 IFCL$<>CHR$(133)THEN900
895 CC=CC+1:PRINT:PRINT:PRINTTAB(15);LEFT$(A$,CC):FORX=1TO500:NEXT
X
896 PRINTCHR$(147):FORX=1TO6:PRINT:NEXTX
898 PRINTB$:PRINT:PRINT
900 INPUT"[SHFT P]REGO RISPONDERE[6 SPZ]";C$
910 GOTO290
1499 REM
1500 FORX=1TO500:NEXT:C=0
1510 PRINTCHR$(147):FORX=1TO6:PRINT:NEXTX
1790 POKES+24,15:POKES+1,130:POKES,0:POKES+5,9:POKES+6,240:POKES+4
,21
1795 PRINT"VERAMENTE OTTIMO!!.....":C=C+1
1797 FORX=1TO600:NEXTX:PRINTCHR$(147)

```

COMMODORE

```

1800 POKES+24,0:POKES+1,0:POKES,0:POKES+5,0:POKES+6,0:POKES+4,0
1810 FORX=1TO6:PRINT:NEXTX
1820 IFC<4GOTO1790
1830 PRINTCHR$(147):RESTORE:FORPR=0TO62:READPQ:NEXTPR
1840 PRINTCHR$(18)" PAROLE SILLABATE OGGI[15 SPZ]"
1850 READA1$
1855 IFA1$="-1"THENEND
1860 READB1$,A2$,B2$
1880 PRINTA1$;:PRINTTAB(20)A2$
1890 GOTO1850
1999 REM
2000 POKES+24,15:POKES+1,45:POKES,0:POKES+5,9:POKES+6,0:POKES+4,17
2010 FORX=1TO400:NEXTX
2020 POKES+24,0:POKES+1,0:POKES,0:POKES+5,0:POKES+6,0:POKES+4,0
2030 RETURN
2499 REM
2500 PRINTCHR$(147)
2510 PRINT:PRINT:PRINT
2520 PRINT"[SHFT L]E PAROLE E LE LORO DEFINIZIONI[2 SPZ]SONO NEI [
SHFT D][SHFT A][SHFT T][SHFT A][4 SPZ]"
2525 PRINT"[SHFT L][SHFT I][SHFT N][2 SHFT E][4 SPZ]....[3 SPZ]299
9-4999"
2530 PRINT:PRINT"[SHFT S]OSTITUITELE COSI':[14 SPZ]"
2540 PRINT:PRINT"LINEE DI [SHFT D][SHFT A][SHFT T][SHFT A] QUANTE
SONO LE PAROLE[2 SPZ]"
2545 PRINT" IN OGNI LINEA PRIMA LA PAROLA[2 SPZ]"
2550 PRINT" POI DI SEGUITO LA DEFINIZIONE "
2555 PRINT:PRINT"OGNI DEFINIZIONE NON DEVE SUPERARE I 39 CARATTERI
[10 SPZ]"
2600 PRINT:PRINTCHR$(18)"[SHFT P]REMERE [SHFT R]ETURN PER VISUALIZ
ZARE LE LINEE";:INPUTF2$
2900 LIST2999-4999
2950 DATA0,0,0,25,191,255,31,191,255,31,156,6,15,12,244,15
2951 DATA45,244,95,252,180,127,252,116,255,252,52,255,255,252,127,
255
2952 DATA254,95,255,255,3,255,255,3,255,255,15,254,0,31,252,254
2953 DATA63,253,131,120,61,57,243,157,57,246,193,131,3,128,254
2999 REM
3000 DATA "CANE","[SHFT C]ORRE DIETRO DA I [SHFT G]IATTI[5 SPZ]"
3002 DATA "PENNA","[SHFT S]I USA PER SCRIVERE[10 SPZ]"
3004 DATA "ELEFANTE","[SHFT H]A LA PROBOSCIDE[10 SPZ]"
3006 DATA "ALFABETO","NE FANNO PARTE A,B,C'S ED ALTRE "
5000 DATA "-1","-1","-1","-1"

```

COMMODORE



FLIPPER



Una simulazione dei FLIPPER da bar realizzata per il Vostro VIC20. Per evitare che la biglia vada persa, uscendo dalla parte sinistra del video, dovete ricorrere ad una paletta, posta anche essa sul lato sinistro, spostabile mediante i tasti di cursore verso l'alto e verso il basso. Buon divertimento.

```

10 PRINT"[ROSS][CLR][4 SPZ]FLIPPER[BLU][GIU']":GOSUB650
20 F=1:G=1
30 PRINT"[CLR][GIA][5 DESTR]*[2 GIU']*[2 SIN.]*[DESTR]*";
40 Y=INT(RND(1)*18)
50 IFY>7ANDY<11THEN40
60 PRINT"[PORP][6 DESTR][GIU']][SHFT Z][2 DESTR][SHFT Z][3 GIU']][3
  SIN.][SHFT Z][2 DESTR][SHFT Z]";
70 PRINT"[GIA][4 GIU']*[2 GIU']][SIN.]*[2 GIU']][SIN.]*";
80 PRINT"[NERO][6 SIN.][3 SU][SHFT U][SHFT I][GIU']][2 SIN.][SHFT
  J][SHFT K]"
90 PRINT"[GIA][5 GIU']][4 DESTR]*[2 GIU']*[2 SU]*";
100 PRINT"[PORP][2 GIU']][6 DESTR][SHFT Z][DESTR][2 SU][SHFT Z][DE
  STR][2 SU][SHFT Z]"
110 PRINT"[HOME][ROSS][DESTR][8 GIU']][SHFT V][GIU']][SIN.][SHFT V]
  [GIU']][SIN.][SHFT V]"
120 FORB=0TO21
130 POKE38840+B,2
140 POKE8120+B,119
150 NEXTB
160 REM MOSSA DEL GIOCATORE
170 POKE38402+P*22,6
180 POKE7682+P*22,117
190 GETA$:GOSUB260
200 IFA$="B"ORA$="S"THEN20
210 PRINT"[HOME][NERO][21 GIU']PUNTI="SC:IFA$=""THEN190
220 POKE7682+P*22,32
230 IFA$="[GIU']"THENP=P+1
  
```

COMMODORE

```

240 IFA$="[SU]"THENP=P-1
250 GOTO160
260 REM MOVIMENTO BIGLIA
270 POKE7680+X+Y*22,32
280 X=X+F:Y=Y+G
290 H=PEEK(7680+X+Y*22)
300 IFH<>32THEN360
310 POKE38400+X+Y*22,5
320 POKE7680+X+Y*22,81
330 IFX=210RX=0THENF=-F
340 IFY=190RY=0THENG=-G
350 RETURN
360 IFH=117THENF=-F:X=X+F:SC=SC+2:N=190
370 IFH=42THENF=-F:X=X+F:SC=SC+20:N=220
380 IFH=90THENG=-G:Y=Y+G:SC=SC+10:N=220
390 IFH=730RH=740RH=750RH=85THENG=-G:Y=Y+G:SC=SC+50:N=220
400 IFH=86THEN440
410 T=50:GOSUB600
420 GOTO310
430 REM BIGLIA
440 Q=Q+1:IFQ=5THEN510
450 N=129:T=200:GOSUB600
460 PRINT"[CLR][2 DESTRI][2 SU]OOOPS!!"
470 PRINT"[RVS ON][2 DESTRI][2 GIU ]BIGLIA "Q"[2 GIU ]15 SIN.1PREM
  . B"
480 GETA$:IFA$<>"B"THEN480
490 RETURN
500 REM
510 N=129:T=300:GOSUB600
520 PRINT"[ROSS][CLR][RVS ON][2 GIU']*****"
530 PRINT"[2 GIU']PUNTI,"SC"*****":IFSC<HSTHEN560
540 GOSUB730
550 HS=SC:PRINT"[GIU']AVETE UNA BIGLIA[6 SPZ]MATT ":GOTO570
560 PRINT"AVETE FATTO ESATTAMENTE PUNTI"HS
570 INPUT"[GIU']ANCORA[3 SPZ](S/N)":A$
580 IFA$="S"THENS=0:Q=0:RETURN
590 END
600 V=36878:S=36874
610 POKEV,9
620 RETURN
630 FORM=1TOT:NEXTM
640 RETURN
650 PRINT"DIFENDETE LA PARTE SINISTRA DEL CAMPO DALLA BIGLIA MATT

```

COMMODORE

```
A *****"
660 PRINT"[GIU']UTILIZZANDO I TASTI DI CURSORE PER MUOVERVI"
670 PRINT"[2 GIU']PUNTEGGI ;[GIU']([SIN.][GIA]*[BLU]=10 PNT[GIU'] [
7 SIN.][FORP][SHFT Z][BLU]=20 PNT"
680 PRINT"[NERO][10 SPZ][GIU'] [2 SIN.][SHFT U][SHFT I][GIU'] [2 SI
N.][SHFT J][SHFT K][BLU]=50 PNT"
690 PRINT"*****"
700 PRINT"PREMI UN TASTO"
710 GETA$:IFA$=""THEN710
720 RETURN
730 V=36878
740 RESTORE
750 FORA=1T012
760 READN,T
770 FORL=1TON
780 POKE36874,T
790 POKEV,8
800 NEXTL
810 POKEV,0
820 NEXTA
830 DATA25,195,25,195,50,209,50,215,25,219,50,219,25,195,25,209,2
5,209,25,215,25,215
840 DATA50,219
850 RETURN
```



DAMA



Per chi si ritrova ad essere appassionato del gioco della dama, un programma da cui avere ampie soddisfazioni che appunto permette di giocare a DAMA. Al lancio, verra' richiesto quanti giocatori hanno intenzione di gareggiare e nel caso venga risposto 0, partira' il dimostrativo, cioe' il computer giochera' contro se stesso. Nel caso non si voglia giocare contro il computer (opzione 1) e possibile anche giocare con un amico, utilizzando la scacchiera come

COMMODORE

una normale scacchiera; in questo caso andranno comunicate le coordinate di partenza e di arrivo di ciascun giocatore. Vi sono alcune regole che valgono sia per l'opzione 1 che per l'opzione 2: Per spostare la pedina dall'attuale punto, a quello in cui si desidera andare, andranno comunicate al computer le coordinate di partenza e quelle di arrivo, sapendo che il primo numero rappresenta le righe ed il secondo le colonne; per "MANGIARE" una pedina ed effettuare in tal caso un salto d'essa andrà preventivamente battuta la lettera J ed andrà comunicato se si tratta di un unico salto o di più salti (caso in cui vengono mangiate più pedine). Il programma non presenta difficoltà rilevanti ed è adatto anche a quanti iniziano ad avvicinarsi per la prima volta alla programmazione.

```

1 RESTORE
5 PRINT"[CLR]":INPUT "NUMERO DI GIOCATORI[3 SPZ](0,1 Q[2 SPZ]2)";
P
10 PRINT"[CLR]":SY(1)=81:SY(2)=102:K=0:I=1:JP(1)=0:JP(2)=0
15 S1=256*PEEK(46)+PEEK(45)+130
18 FORZ=0TO127:READB%:POKE51+Z,B%:NEXT
20 SYSS1
25 PRINT"[HOME]";"[CYN]";"[5 SPZ]COL 1[3 SPZ]2[3 SPZ]3[3 SPZ]4[3
SPZ]5[3 SPZ]6[3 SPZ]7[3 SPZ]8";
30 FORRO=1TO8:PRINT:PRINT:PRINT"[6 SPZ]";RO:NEXT
35 PRINT"[HOME]";PRINT:PRINT"[5 SPZ]R":PRINT"[5 SPZ]I":PRINT"[5 S
PZ]6"
40 PRINT"[4 GIU'] [BLU]SALTO[3 GIU'] [4 SIN.] DA [3 GIU'] [3 SIN.] [G
IA]A [GIU']";
45 FORRO=1TO3:READLS
50 FORCO=1TO7STEP2
55 L=LS+4*CO
60 POKEL,SY(I):NEXTCO:NEXTRO
65 I=I+1:IFI<3THEN45
70 POKE55698,6:POKE55699,6:POKE55818,9:POKE55819,9:POKE55938,7
75 POKE55939,7:POKE55978,7:POKE55979,7:POKE56018,7:POKE56019,7
80 POKE56058,7:POKE56059,7:POKE56098,7:POKE56099,7:POKE55578,0
85 POKE56258,2:POKE56259,2:POKE56260,2
90 I=2:J=0:IFP=0THEN400
95 POKE1306,SY(I)

```

COMMODORE

```

100 GETL$: IFL$="" THEN 100
105 IFL$(">"J" THEN R$=L$: GOTO 135
110 POKE 1426, 10
115 GETJ$: IFJ$="" THEN 115
120 IF ((ASC(J$)-53>0) OR (ASC(J$)-49<0)) THEN 115
125 POKE 1427, ASC(J$): J=ASC(J$)-48
130 GETR$: IFR$="" THEN 130
135 IF ((ASC(R$)-56>0) OR (ASC(R$)-49<0)) THEN 100
140 POKE 1546, ASC(R$)
145 GETC$: IFC$="" THEN 145
150 IF ((ASC(C$)-56>0) OR (ASC(C$)-49<0)) THEN 145
155 POKE 1547, ASC(C$)
160 GETR2$: IFR2$="" THEN 160
165 IFR2$="C" THEN 305
170 IF ((ASC(R2$)-56>0) OR (ASC(R2$)-49<0)) THEN 160
175 POKE 1666, ASC(R2$)
180 GETC2$: IFC2$="" THEN 180
185 IF ((ASC(C2$)-56>0) OR (ASC(C2$)-49<0)) THEN 180
190 POKE 1667, ASC(C2$)
195 IFJ>1 THEN 800
200 POKE 1986, 15: POKE 1987, 11: POKE 1988, 63
205 GETAN$: IFAN$="" THEN 205
210 IFAN$(">"S" THEN 310
215 L=989+120*(ASC(R$)-48)+4*(ASC(C$)-48)
220 L2=989+120*(ASC(R2$)-48)+4*(ASC(C2$)-48)
225 IF PEEK(L+1)(">"160 THEN K=1
230 DR=ASC(R2$)-ASC(R$): DC=ASC(C2$)-ASC(C$): IF PEEK(L)(">"SY(I) THEN 3
10
235 IF PEEK(L2)(">"160 THEN 310
240 IF ((ABS(DR)>2) OR (ABS(DR)<1)) THEN 310
245 IF ((ABS(DR)=2) AND (L$(">"J")) THEN 310
250 IF ((ABS(DR)=1) AND (L$="J")) THEN 310
255 IF ABS(DC)(">"ABS(DR) THEN 310
260 IF ((I=1) AND (K=0) AND (DR<0)) THEN 310
265 IF ((I=2) AND (K=0) AND (DR>0)) THEN 310
270 IF ((ABS(DR)=2) AND ((PEEK((L+L2)/2)=SY(I)) OR (PEEK((L+L2)/2)=160
))) THEN 310
275 IF ((R2$="1") OR (R2$="8")) THEN K=1
280 POKE L, 160: POKE L+1, 160
285 FOR NK=0 TO K: POKE L2+NK, SY(I): NEXT
290 IF ABS(DR)=2 THEN POKE (L+L2)/2, 160: POKE (L+L2)/2+1, 160: JP(I)=JP(I
)+1
292 IF JP(I)=12 THEN 600

```

COMMODORE

```

295 IFJ<2THEN305
300 GOTO900
305 K=0:JF=0:I=I+1:IFI=3THENI=1
310 POKE1986,32:POKE1987,32:POKE1988,32
315 IFJF=1THENPOKE1666,160:GOTO160
320 POKE1426,32:POKE1427,32:POKE1546,32:POKE1547,32:POKE1666,32
325 POKE1667,32:POKE1706,32:POKE1707,32:POKE1746,32:POKE1747,32
330 POKE1786,32:POKE1787,32:POKE1826,32:POKE1827,32:J=0
335 IF((P=1)AND(I=1))THEN400
340 GOTO95
400 NT=0:R=INT(8*RND(1)+1):C=INT(8*RND(1)+1):POKE1306,SY(I)
405 L=989+120*R+4*C:IFPEEK(L)<>SY(I)THEN400
410 NT=NT+1:IFNT>10THEN400
415 DR=INT(3*RND(1)-1):IFDR=0THEN415
420 SN=INT(2*RND(1)):DC=DR
425 IFSN=0THENDC=-DC
430 L2=989+120*(R+DR)+4*(C+DC)
435 IFPEEK(L2)=SY((NOTI)+4)THENL2=L2+120*DR+4*DC:DR=2*DR:DC=2*DC
440 IFPEEK(L2)<>160THEN410
445 IF((ABS(DR)=2)AND((PEEK((L+L2)/2)=SY(I))OR(PEEK((L+L2)/2)=160
)))THEN410
450 IFPEEK(L+1)<>160THENK=1
455 IF((I=1)AND(K=0)AND(DR<0))THEN410
460 IF((I=2)AND(K=0)AND(DR>0))THEN410
465 IF((R+DR=1)OR(R+DR=8))THENK=1
470 POKEL,160:POKEL+1,160
475 FORNK=0TOK:POKEL2+NK,SY(I):NEXT
480 IFABS(DR)=2THENPOKE(L+L2)/2,160:POKE(L+L2)/2+1,160:JP(I)=JP(I
)+1
485 IFJP(I)=12THEN600
490 K=0:I=I+1:IFI=3THENI=1
495 IFP=0THEN400
500 GOTO95
600 PRINT"[CLR]":S=54272:POKES+24,15:POKES+5,9:POKES+6,0:POKE5328
1,0
605 FORN=0TO5:READH,L,D
610 POKES,L:POKES+1,H:POKES+4,33
615 FORT=1TOD:NEXTT
620 POKES+4,32:FORT=1TO10:NEXTT:NEXTN
625 FORN=1TO20:POKE1080+40*N,SY(I):POKE1082+40*N,23:POKE1083+40*N
,9
630 POKE1084+40*N,14:POKE1085+40*N,19:NEXTN
635 END

```

COMMODORE

```

800 FORM=2TQJ
805 POKE1626+40*M,160
810 GETR3$(M):IFR3$(M)=""THEN810
815 IF((ASC(R3$(M))-56>0)OR(ASC(R3$(M))-49<0))THEN810
820 POKE1626+40*M,ASC(R3$(M))
825 POKE1627+40*M,160
830 GETC3$(M):IFC3$(M)=""THEN830
835 IF((ASC(C3$(M))-56>0)OR(ASC(C3$(M))-49<0))THEN830
840 POKE1627+40*M,ASC(C3$(M))
845 NEXTM:GOTO200
900 J=J-1:POKE1427,J+48:JF=1
905 POKE1546,PEEK(1666):POKE1547,PEEK(1667)
910 FORL=0TO4:POKE1666+L*40,PEEK(1706+L*40)
915 POKE1667+L*40,PEEK(1707+L*40):NEXT
920 R$=CHR$(PEEK(1546)):C$=CHR$(PEEK(1547))
925 R2$=CHR$(PEEK(1666)):C2$=CHR$(PEEK(1667))
930 GOTO215
1020 DATA169,12,141,32,208,141,33,208,162,0,142,254,3,142,255,3
1030 DATA142,252,3,162,48,134,251,134,253,162,2,142,253,3,162,4
1040 DATA134,252,162,216,134,254,160,0,169,160,145,251,152,74,74,
41
1050 DATA1,170,189,252,3,145,253,200,192,32,208,236,238,255,3,173
1060 DATA255,3,201,24,208,1,96,238,254,3,173,254,3,201,3,208
1070 DATA19,173,252,3,72,173,253,3,141,252,3,104,141,253,3,169
1080 DATA0,141,254,3,169,40,24,101,251,133,251,133,253,144,183,16
9
1090 DATA1,24,101,252,133,252,169,1,24,101,254,133,254,144,167,96
1100 DATA1109,1233,1349,1713,1829,1953
1110 DATA16,195,64,21,31,64,25,30,64,33,135,128,25,30,64,33,135,1
28
3000 REM -----

```



SUPERSINT



Un autentico ausilio per gli amanti della MUSICA ELETTRONICA che trasformerà il COMMODORE 64 in un vero SINTETIZZATORE. Le istruzioni sono contenute nel programma. Per sospendere il programma dovrete

COMMODORE

premere la BARRA SPAZIATRICE.
buone COMPOSIZIONI.

Buon divertimento e

```

5 POKE53280,0:POKE53281,0
10 PRINT"[CLR]"
15 PRINT"[RVS ON][40 SPZ][RVS OFF]";
20 PRINT"[RVS ON][3 SPZ]IL SINTETIZZATORE COMMODORE 64[7 SPZ]
[RVS OFF]";
25 PRINT"[RVS ON][40 SPZ][RVS OFF]";
30 PRINT:PRINT"ESSO UTILIZZA UNA VOCE PER SUONARE[4 SPZ]"
40 PRINT"SU 3 OTTAVE DI RANGE"
50 PRINT"LA FORMA D'ONDA E' MODIFICABILE MEDIANTE"
53 PRINT" [RVS ON]F1[RVS OFF] X LA TRIANG. ,[RVS ON]F3[RVS OF
F] X LA DENTE S.";
56 PRINT"[RVS ON]F5[RVS OFF] X LA QUADRA[4 SPZ][RVS ON]F7[RVS
OFF] X IL RUM."
58 PRINT"SONO RICHIESTI I PARAM. ADSR ED OND.Q."
60 PRINT"-----"
70 PRINT"NOTE ACUTE SONO I TASTI 2,3,4,6,7,9,0,+,";
75 PRINT"[SHFT M] E[3 SPZ][RVS ON]CLR/HOME[RVS OFF]"
80 PRINT:PRINT"LE NOTE NORMALI SONO DA[2 SPZ]Z A[2 SPZ]/[2 SP
Z]E "
90 PRINT"DA[3 SPZ]Q A[2 SPZ]^"
95 PRINT:PRINT"[6 SPZ][RVS ON] PREMI UN TASTO[7 SPZ][RVS OFF]
"
96 GETA$:IFA$=""THEN96
98 PRINT"[CLR]"
100 DIMNO(64,2)
110 FORC=1TO62
120 READNO(C,1),NO(C,2)
130 NEXTC
140 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
150 DATA27,56,25,177,0,0,30,141,8,147,9,21,28,214
160 DATA0,0,0,0,32,94,10,60,36,85,10,205
170 DATA0,0,34,75,9,159,40,200,38,126,12,32
180 DATA0,0,12,216,13,156,43,52,11,114,48,127
190 DATA45,198,15,70,54,111,16,47,0,0,51,97
200 DATA14,107,61,126,57,172,18,42,0,0
210 DATA19,63,20,100,64,88,17,37,72,169,68,149
220 DATA0,0,81,161,0,0,0,0,76,252,21,154
230 DATA0,0,0,0,0,0,24,63,0,0,0,0,22,227,0,0
240 VOL=54296

```

COMMODORE

```

250 HFR=54273
260 LFR=54272
270 WFM=54276
280 HPE=54275
290 LPE=54274
300 AD=54277
310 SR=54278
320 PRINT"FORME D'ONDA: "
330 PRINT"17 = TRIANGOLARE"
340 PRINT"33 = DENTE DI SEGA"
350 PRINT"65 = QUADRA"
360 PRINT"129= RUMORE"
365 INPUTR
370 INPUT"ATTACCO/DECADIMENTO";L(1)
380 INPUT"SOSTENTAMENTO/RILASCIO";L(2)
390 IFR<>65THEN420
400 INPUT"QUADRA ALTA (0-15)";P(1)
410 INPUT"QUADRA BASSA[2 SPZ](0-255)";P(2)
420 REM
430 POKEVOL,15
435 POKELPE,P(2):POKEHPE,P(1)
440 POKEAD,L(1)
450 POKESR,L(2)
455 POKEWFM,R
460 S=PEEK(197)
462 POKE198,0
463 GOSUB2000
465 IFS=60THENPOKEVOL,0:POKEWFM,0:POKEAD,0:POKESR,0:GOTO9999
490 IFS=64THEN460
500 POKEHFR,NO(S,1)
510 POKELFR,NO(S,2)
515 IFPEEK(197)=STHEN515
520 POKEWFM,0
540 GOTO430
600 POKEVOL,5
610 GOTO430
2000 IFS=4THENR=17
2010 IFS=5THENR=33
2020 IFS=6THENR=65:GOTO400
2030 IFS=3THENR=129
2040 RETURN
5200 POKEAD,0:POKESR,0:POKEWFM,0
9999 PRINT"[CLR]"

```

COMMODORE



DISEGNO



Avete mai provato a disegnare sul video del Vostro VIC20, utilizzando il JOYSTICK?... Questo programma Ve ne offre la possibilità. Utilizzando infatti il JOYSTICK, potrete realizzare fantasiosi disegni direttamente sul video. Nel caso vogliate spostarVi senza disegnare, oppure cancellare una linea appena tracciata, utilizzate il pulsante di FUOCO del JOYSTICK. Non temete di andare oltre la cornice visualizzata sul video, perche' il programma gestisce anche questo, facendoVi rientrare da un'altra parte.

```
0 POKE36879,8:POKE36867,21:POKE36864,17:POKE36865,45:POKE36866,14
4:POKE36869,253
1 PRINT"[CLR]":FORA=1TO16:FORN=1TO10:POKE7663+16*N+A,N+10*A-11:NE
XTN,A
2 FORA=5120TO7679:POKEA,0:NEXT:POKE37154,127:X=64:Y=80
3 FORA=5120TO7679STEP160:POKEA,255:POKEA+159,255:NEXT
4 FORA=5121TO5278:POKEA,128:POKEA+2400,1:NEXT
10 A=PEEK(37137):X=X+((AAND16)=0):Y=Y+((AAND4)=0)
15 Y=Y-((AAND8)=0):X=X-((PEEK(37152)AND128)=0)
20 F=-((AAND32)=0):IFF=1THEN60
50 Y1=Y+5120+160*INT(X/8):POKEY1,PEEK(Y1)OR2^(7-XAND7):GOTO10
60 Y1=Y+5120+160*INT(X/8):POKEY1,PEEK(Y1)ANDNOT2^(7-XAND7):GOTO10
```

DIDATTICA.DIDATTICA.DII

Quando cominciano ad essere sviluppati programmi di una certa consistenza uno dei problemi maggiori che si

COMMODORE

incontrano e quello della visualizzazione dei dati, particolarmente di quelli numerici. I computer sono nati e vengono per la maggior parte prodotti in INGHILTERRA ed AMERICA ed il caso vuole che in questi paesi la nostra virgola decimale, utilizzata per separare i decimali dalla parte intera del numero e' sostituita dal punto. Risultato: numeri che a prima vista, nel caso non si conosca questo scambio, risultano incomprensibili. Il programma didattico di cui ci occuperemo questa volta risulta quindi essere anche una autentica UTILITY per quanti progettano programmi di tipo gestionale. Ecco quindi il programma e di seguito la spiegazione accurata.

```

10 Z$ = "0000000000"
20 HOME
30 V = 20
40 D = 3
50 INPUT R
60 PRINT "CIFRE";
70 REM -----
80 N$ = STR$ (R)
90 L = LEN (N$)
100 C = 0
110 C = C + 1
120 IF L < C THEN GOTO 150
130 X$ = MID$ (N$,C,1)
140 IF X$ < > "." THEN 110
150 L1 = C - 1
160 L2 = L - C
170 IF L1 = 0 THEN GOTO 210
180 PRINT TAB( V - L1);
190 PRINT LEFT$ (N$,L1);
200 GOTO 220
210 PRINT TAB( V - 1);"0";
220 IF L2 < = 0 THEN 290
230 PRINT ",";
240 IF L2 < D THEN 270
250 PRINT MID$ (N$,C + 1,D);

```

COMMODORE

```

260 GOTO 290
270 PRINT MID$(N$,C + 1,L2);
280 PRINT LEFT$(Z$,D - L2);
290 REM -----
300 PRINT
310 GOTO 50

```

.*****Istruzioni 10-70: INIZIALIZZAZIONE *** Vengono caricati degli zeri nella variabile di tipo alfanumerico Z\$. Essendo la variabile di tipo alfanumerico sono necessarie le virgolette (") all'inizio ed alla fine degli zeri. Essi serviranno a completare la parte intera del numero e cioè quella prima della virgola. Viene cancellato il video mediante la parola chiave HOME (CLS per lo SPECTRUM, CLR per i COMMODORE, CALL CLEAR per il TI99/4A della TEXAS). Nella variabile numerica V viene caricato il valore 20, corrispondente alla posizione sul video della virgola. Nella variabile numerica D viene caricato il valore 3, corrispondente al numero dei decimali presenti dopo la virgola. Viene richiesto di immettere il numero da convertire in maniera comprensibile a noi Italiani (mediante la parola chiave INPUT). Viene creato una divisione logica nel listato del programma mediante la parola chiave REM ed alcuni trattini. Istruzioni 80-100: TRATTAMENTO DATI ***** La variabile numerica R, contenente il nostro numero viene convertita mediante la parola chiave STR\$ nella variabile di tipo alfanumerico N\$. Mediante la parola chiave LEN viene determinata la lunghezza sempre del nostro numero, caricandola nella variabile numerica L. Viene azzerata la variabile numerica C che servirà a contenere la posizione del punto (nella notazione numerica ANGLOSASSONE) decimale all'interno del numero. Istruzioni 110-160: RICERCA DEL PUNTO DECIMALE ***** Viene effettuata la ricerca del punto decimale all'interno del nostro numero. A questo scopo viene di volta in volta incrementata di uno la variabile numerica C. Viene controllato se essa è superiore alla lunghezza del numero ed in tal caso salta all'istruzione 150. La parola chiave MID\$ serve ad estrarre dei caratteri da una variabile

COMMODORE

alfanumerica ed in questo caso la variabile N\$. Il carattere C presente di seguito dopo la virgola e' una variabile e per l'esattezza la variabile che ci indica la posizione del punto decimale all'interno del numero. Il numero 1 indica che vogliamo estrarre solo un carattere a partire dalla posizione C (il numero contenuto in essa). Questa istruzione serve per estrarre dal numero il punto decimale nel caso che il confronto effettuato all'istruzione 140 non si verifichi. Nel caso si tratti effettivamente di un punto decimale viene caricato nella variabile L1 il numero di caratteri che precedono la virgola, mentre nella variabile L2 il numero di caratteri che seguono la virgola (il punto).

Istruzione 170: Viene controllato se la variabile L1 e' uguale a zero (caso di numero senza decimali) ed in caso affermativo viene effettuato un salto all'istruzione 210.

Istruzioni 180-200: STAMPA PARTE INTERA ***** Mediante la parola chiave TAB viene posizionato il cursore sul video alla posizione della virgola e mediante la parola chiave LEFT\$ viene stampata la parte intera del numero e fatto questo viene effettuato un salto all'istruzione 220.

Istruzione 210-220: Servono a stampare i numeri senza decimali e ad andare alla fine del programma.

Istruzioni 230-280: Servono a stampare la virgola alla posizione prefissata e la parte di decimali massimi confrontandoli con il valore della variabile numerica D dell'istruzione 40. Istruzioni 290-310: servono a far tornare il programma all'inizio pronto a ricevere un nuovo numero da convertire e visualizzare.

~~~~~ ERRORE ~~~~~

Purtroppo cari lettori, a maneggiare listati e commenti si finisce per commettere qualche errore. Nel numero scorso infatti, a pagina 9, il programma "Bilancio Familiare" si e' presso il commento di un altro programma. Eccovi quindi, e ci scusiamo, il commento "giusto" da abbinare al listato pubblicato.

# COMMODORE

Nella nostra vita quotidiana uno degli aspetti forse piu' importanti e' la quadratura del cosidetto "BILANCIO FAMILIARE", cosa non semplice dati i prezzi dei generi necessari al nostro fabbisogno quotidiano. Quindi questo programma per il Vostro amato COMMODORE 64 capita veramente a proposito. Anche se a prima vista potra' sembrare lungo e complesso non Vi scoraggiate perche' le apparenze ingannano. La prima cosa che dovrete fare prima di iniziare a caricarlo sara' quella di battere il seguente comando: PRINT CHR\$(14). Esso serve a convertire i caratteri da maiuscolo a minuscolo. Le lettere maiuscole presenti nel listato si ottengono premendo il carattere interessato unitamente al tasto SHIFT. Ma veniamo all'utilizzazione del programma. La prima cosa che Vi sara' richiesta sara' il nome del FILE su cui intendete conservare i dati del vostro bilancio familiare. Fatto questo Vi apparira' il MENU' PRINCIPALE, mediante il quale potrete: con l'opzione 1 immettere i vari dati del Vostro bilancio ad esempio le entrate o le spese varie, mediante l'opzione 2 potrete variare dei dati erroneamente caricati, con l'opzione 3 potrete visualizzare o stampare la totalita' del vostro bilancio, con l'opzione 4 potrete controllare quanti soldi Vi rimangono, con l'opzione 5 potrete analizzare le varie voci delle Vostre spese ad esempio gli abiti o la spesa giornaliera ed oltretutto controllare in quale percentuale essi incidono sulle vostre spese. Soffermiamoci ora sull'immissione ed analizziamo cosa immettere nei vari campi: nel primo deve essere immessa la data del giorno, nel secondo deve essere immesso il codice del tipo ad esempio ent per entrate oppure lib per libri oppure var per varie ecc., nel terzo campo deve essere immessa la descrizione ad esempio nel caso di entrate (codice ent) potra' essere battuto stipendio mensile, nel quarto campo deve essere immesso il numero di verifica progressiva effettuata, nel quinto campo deve essere immesso l'importo della spesa fatta (attenzione questo campo deve essere lasciato in bianco nel caso si tratti di entrate come ad esempio lo stipendio), nel sesto campo deve essere immesso l'importo dell'entrata ad esempio lo stipendio (attenzione quando viene

# COMMODORE

impresso nel caso di entrate qualcosa in questo campo, nulla deve essere impresso nel precedente campo riguardante le uscite). Buon lavoro!....

Un'ultima cosa, per uscire dall'immissione o dalla variazione basterà battere la parola fine nel campo della data e premere il tasto RETURN.



## COMMODORE MIX



### COME FUNZIONA LA VARIABILE DI STATUS

di Renato Gualtieri

Forse, prima di cominciare a discutere sul come funziona questa variabile Particolare del C-64 e' bene dire piu' in generale cos'e' la variabile di stato (o ST, nella forma usata in BASIC). Quando si effettua una operazione su un dispositivo fisico, sia esso l'unita' a cassette, l'unita' a dischetti, il video, la tastiera, e tutto quello che puo' essere attaccato alla porta RS-232 e IEEE488, il dispositivo interessato segnala in qualche maniera al computer come e' andata l'operazione.

Come lo fa? Molto semplice: se e' un dispositivo attaccato alla porta seriale RS232 lo scrive alla locazione esadecimale 297, se e' un qualsiasi altro dispositivo lo scrive all'indirizzo esadecimale 90. Non preoccupatevi pero', perche' non e' necessario andare a vedere queste posizioni di memoria per sapere com'e' andata l'operazione. Il C-64 infatti, vi mette il valore registrato nelle locazioni dette in una variabile riservata, la variabile ST (Status). Questo fatto e' molto comodo, infatti per sapere come e' andata l'operazione appena effettuata non occorre fare delle strane PEEK o POKE, ma bastera' fare una semplice 'PRINT ST'. Che valori potrete trovarci in questa ST? Dunque, la ST e' lunga un byte e di questo byte viene usato un solo bit per volta, per cui i valori possibili sono: 1, 2, 4, 8, 16, 32, 64, e -128. Se date subito un'occhiata alle tabelle che seguono vedrete il significato di ogni valore in relazione al dispositivo che ha dato la segnalazione.

# COMMODORE

| STATO    | DISPOSITIVI                 |                              |                                      |                                      |
|----------|-----------------------------|------------------------------|--------------------------------------|--------------------------------------|
|          | !Lettura<br>!da<br>!casset. | !Verifica<br>!da<br>!casset. | !Unità<br>!a<br>!dischi              | !IEEE488                             |
| 00000001 | !Operaz.<br>!OK             | !Operaz.<br>!OK              | !Unità<br>!non<br>!Pronta<br>!(lett) | !Unità<br>!non<br>!Pronta<br>!(lett) |
| 1        |                             |                              |                                      |                                      |
| 00000010 | !Operaz.<br>!OK             | !Operaz.<br>!OK              | !Unità<br>!non<br>!Pronta<br>!(scr.) | !Unità<br>!non<br>!Pronta<br>!(scr.) |
| 2        |                             |                              |                                      |                                      |
| 00000100 | !Blocco<br>!dati<br>!corto  | !Blocco<br>!dati<br>!corto   |                                      |                                      |
| 4        |                             |                              |                                      |                                      |
| 00001000 | !Blocco<br>!dati<br>!lungo  | !Blocco<br>!dati<br>!lungo   |                                      |                                      |
| 8        |                             |                              |                                      |                                      |
| 00010000 | !Errore<br>!di<br>!lettura  | !Errore<br>!di<br>!verifica  |                                      |                                      |
| 16       |                             |                              |                                      |                                      |
| 00100000 | !Errore di Parità           |                              | !Uno o più bit<br>!letti male        |                                      |
| 32       |                             |                              |                                      |                                      |
| 01000000 | !Fine<br>!del<br>!file      |                              | !Fine<br>!del<br>!file               | !Fine<br>!della<br>!identifica       |
| 64       |                             |                              |                                      |                                      |
| 10000000 | !Fine<br>!del<br>!nastro    | !Fine<br>!del<br>!nastro     | !Unità<br>!non<br>!colleg            | !Unità<br>!non<br>!Presente          |
| -128     |                             |                              |                                      |                                      |

Se il valore che vi trovate non è nella tabella, vuol dire che le condizioni rilevate sono più di una, ma anche in questo caso, con un minimo d'attenzione, si può agevolmente risalire a quanto successo. Infatti, se i valori usati e possibili sono quelli che ho indicato, altri valori che potreste trovarvi saranno il frutto di una somma algebrica dei valori succitati, e ci sarà una sola combinazione che darà il risultato che voi

# COMMODORE

avete. Per essere piu' chiari: se la ST contenesse il valore '104', l'unico modo per avere questo numero, usando i valori possibili della variabile, e'  $2+16-128$ , per cui siete davanti alle condizioni rilevate dai valori 2, 16, e -128. Semplice no?

## Fine del File sul Datassette

Se state usando il registratore a cassette in lettura, la variabile ST avra' il valore 64 quando il C-64 si accorgera' che il file che sta leggendo e' finito. In pratica, per andare un pochino piu' in dettaglio, il C-64 ha letto un byte a zero nel file. Questo vuol dire che se nel file o nei vostri dati ci sono dei byte a zero, l'indicatore di fine file (End of File) sara' attivato. I dati sono scritti su nastro in blocchi di 191 byte, ed un byte a zero e' aggiunto quando tutto il file e' stato scritto (ossia quando viene fatta la CLOSE per i dati oppure il programma e' finito). Ma non preoccupatevi eccessivamente ora. Non sto dicendo che se c'e' un byte a zero nei vostri programmi il C-64 si ferma, sto attirando l'attenzione sul fatto che se voi vi apprestate ad usare dei test sulla variabile ST nei vostri programmi, potrete trovarvi una indicazione che non corrisponde alla realta' dei fatti. Comunque e' molto difficile scrivere un byte a zero, se non con la CHR\$(0).

Facciamo un piccolo esperimento. Provate ad immettere ed a far partire il listato che segue:

READY.

```
10 REM PROVA SCRITTURA BYTE A ZERO
20 OPEN 1,1,2,"PROVA"
30 PRINT#1,"A";CHR$(0);"B"
40 PRINT#1,"C":CLOSE1
50 PRINT "RIPOSIZIONARE IL NASTRO"
60 PRINT "E PREMERE RETURN QUANDO PRONTI"
70 INPUT X$
80 OPEN 1,1,0,"PROVA"
90 PRINT "DATO    LUNGHEZZA    ST"
100 INPUT#1,X$:PRINT " ";X$,LEN(X$),ST
110 INPUT#1,X$:PRINT " ";X$,LEN(X$),ST
120 CLOSE1
```

READY

Come vedete le linee 30-40 scrivono due linee su un file su nastro, le linee 50-70 vi chiedono di riavvolgere il nastro e di farlo ripartire, per fare si' che le linee 100-120 possano rileggere il nastro. Facendo partire il programma, otterrete, su

# COMMODORE

video, il seguente risultato:

DATO LUNGHEZZA ST

A 1 64

C 1 64

C'è qualcosa che non quadra. La seconda stringa letta è C, il che implica che B è stato letto come parte della stringa precedente, nonostante ci fosse un byte a zero tra la A e la B. D'altro canto B non lo vediamo come parte di A, che, come ci viene stampato è lungo un byte solamente. Inoltre ci è stato dato un indicatore di fine file (ricordate? È il 64) dopo la stringa A. Dov'è finito B? .... Forse non è il caso di scomodare il tenente Colombo per saperlo. L'istruzione INPUT# legge i suoi dati in una memoria di transito (input buffer).

Quando questi dati sono messi in una variabile di stringa (nel nostro caso la X\$), i dati devono essere mossi in un'altra area di memoria, dove i caratteri delle stringhe sono memorizzati e gestiti. Nel processo che sposta i dati dall'input buffer alla memoria delle stringhe, viene usato un byte a zero per indicare la fine dei dati che devono essere memorizzati. La presenza di uno zero in mezzo ai dati comporta l'arresto del processo di spostamento prematuramente. Per cui: B è stato letto nel buffer di INPUT, ma non è stato poi piazzato nella memoria delle stringhe. Anche in questo caso, come quando si legge da nastro, una volta indicata la fine dei dati, il processo di trasferimento termina, anche se la lettura dei dati può continuare. Nel nostro programmino, la PRINT# fa in modo che dopo la C venga salvato un ritorno carrello, in modo tale da far scrivere i dati in maniera corretta.

## Errori nella lunghezza dei blocchi

Un BLOCCO DATI CORTO è indicato dal valore 4 nella variabile ST (o da un 4 facente parte della somma algebrica che fornisce il risultato in ST). Un BLOCCO DATI LUNGO è identificato da un 8 nelle stesse circostanze. Questi indicatori significano che il blocco di dati letto da nastro non è lungo 191 byte come ci si aspettava, ma ha una lunghezza diversa. E' senza dubbio una condizione di errore. Qualche dato si è "intrufolato" a nostra insaputa, o qualcuno dei nostri dati se ne è andato alla chetichella.

C'è però un'altra eventualità. Se stiamo leggendo un file contenente un programma, come se fosse un file di dati, potremmo, con una probabilità del 99,9%, beccarci questi errori. Il motivo è presto detto: i programmi sono salvati su nastro in un blocco

# COMMODORE

solo e in due copie (ossia in due blocchi in tutto). La seconda copia e' usata Per vedere se ci sono errori nel Primo blocco. E' estremamente improbabile che un Programma occupi esattamente 191 byte; questo il motivo degli errori. Se non ci credete Provate il seguente Programma. Il risultato sara'

```
1      4
```

READY.

```
20 SAVE"PROVA"
50 PRINT "RIPOSIZIONARE IL NASTRO"
60 PRINT "E PREMERE RETURN QUANDO PRONTI"
70 INPUT X$
80 OPEN 1,1,0,"PROVA"
100 GET#1,X$:PRINT LEN(X$),ST
120 CLOSE1
```

READY.

Il 4 stampato indica che il nostro Programma e' Piu' corto dei fatidici 191 byte. Se vogliamo ottenere l'errore di BLOCCO LUNGO Potremmo tentare di allungare il Programma con qualche REM:

```
10 REM PROVA DI LETTURA DI PROGRAMMA
15 REM COME SE FOSSE FILE
```

Se fate girare ora il Programma avrete un bel:

```
1      32
```

Che non e' quello che esattamente ci si aspettava. Il 32 nella ST indica un errore di Parita' ( uno o Piu' bit sono stati letti in maniera non corretta). Parlando sempre di file, con ogni blocco di 191 byte ci sara' scritto quello che viene chiamato "Checksum" e cioe' la somma di tutti i byte. Questa somma viene fatta Per essere sicuri che i dati letti siano stati letti correttamente. Dopo la lettura di ogni blocco viene fatto il test su questa somma. Solo nel caso, fortuito, che il nostro Programma avesse un byte che fosse identico al Checksum calcolato dal C-64, avremmo avuto il messaggio di BLOCCO TROPPO LUNGO.

Fine nastro

La fine del nastro viene indicata con un -128 nella ST. Il Programma che segue vi da' una Piccola dimostrazione:

# COMMODORE

READY.

```
10 OPEN 1,1,2,"PROVA"
20 PRINT#1,"A":CLOSE1
50 PRINT "RIPOSIZIONARE IL NASTRO"
60 PRINT "E PREMERE RETURN QUANDO PRONTI"
70 INPUT X$
80 OPEN 1,1,0,"NIENTE"
100 PRINT ST
120 CLOSE1
```

READY.

Direi che ci Prova,Perche' come avete Potuto vedere il risultato,anche in questo caso,non e'quello che ci si aspetterebbe dalla teoria.Non ci sono Possibilita'di riuscire a "vedere" il fantomatico -128.Il messa99io che vi da' il Programma e' infatti:

"?DEVICE NOT PRESENT ERROR IN 80".

Piu'avanti vedremo che quando si sta usando il bus seriale,un -128 nella ST indica che e' stato fatto un tentativo di mandare dei dati ad una unita'non collegata al computer (come dice il messa99io).

Quando il BASIC si becca un -128 nella ST mentre c'e'un file aperto in lettura,abortisce il Programma con il messa99io di cui sopra, senza controllare se l'unita'coinvolta e' il Datassette, nel cui caso il -128 era dovuto alla rilevazione di fine nastro,oppure una Periferica del bus seriale,come il dischetto o il modem,nel cui caso il -128 indica appunto la condizione di unita'non collegata.Dato che questo messa99io che il C-64 ci manda erroneamente ci butta fuori dal Programma,non c'e'Possibilita'reale di "vedere" da vicino il -128 con l'unita'a nastro.Da notare che questo messa99io lo becchiamo solo noi Poveri tapini,in quanto il BASIC non si pone di queste restrizioni:Lui fa il controllo,durante il LOAD o la VERIFY,per vedere se ha letto l'ultimo Programma su nastro.

Un'altra condizione Possibile della ST,che riguarda le operazioni sul Datassette,e' il valore 16 che significa "Errore di lettura".In Pratica uno o Piu' byte non sono stati letti.Anche in questo caso Pero'ci sara' un errore di Checksum ed e' Proprio quello che Probabilmente di beccheremo.A Proposito del 16,c'e'da dire che il C-64 quando legge un Programma da nastro,verifica,dopo la Prima copia,se ci sono stati dei byte letti male,appunto verificando se c'e'un 16 compreso nel valore della ST.Se il BASIC si accorge che ci sono stati troppi byte letti male,usa la seconda copia del Programma per caricare il tutto. Questo ci garantisce una buona sicurezza dei nostri

# COMMODORE

Programmi salvati su nastro. Il temuto "LOAD ERROR" non si vede spesso sul C-64.

## Il Bus Seriale

Il Bus Seriale è rappresentato esternamente da un connettore circolare a cui si può attaccare l'unità a dischetti, il Plotter, la stampante etc. Nella tabella che segue potete vedere i vari valori di ST ed il loro significato.

Cominciamo a vedere la fine della identificazione (End of Identify). Questo indicatore, valore 64, indica come l'EOF sull'unità a nastro che è stata raggiunta la fine dei dati, mentre si era in lettura. Il punto importante da chiarire è se questa indicazione ce la becchiamo mentre leggiamo l'ultimo byte dei dati o ci viene segnalata quando procediamo con la lettura dopo l'ultimo byte. Una semplice prova vi rivelerebbe che, come l'EOF su nastro, anche l'EOI sulla porta seriale ci viene dato con l'ultimo byte letto. L'unica (importante) differenza è che l'unità a dischetti è in grado di sapere con più esattezza quando ha mandato l'ultimo byte di dati: non c'è neppure un pericolo ad usare dei byte a zero. Tutto ciò significa anche che l'unità a dischetti fa qualcosa di diverso dall'unità a nastro. Diciamo subito che l'indicatore EOI rimane attivo anche se si provvede nella lettura. Inoltre, sempre se si procedesse nella lettura, viene attivato anche l'indicatore di "Dispositivo trasmettitore non disponibile" e cioè si avrà la ST che conterra un 66 ( $64 + 2$ ). Direi che le cose vanno decisamente meglio che non sull'unità a nastro.

## Indicatore di unità non presente

Questo indicatore viene segnalato mettendo un -128 nella somma della ST. In pratica vuol dire che avete provato a far fare qualcosa ad una unità e questa non era fisicamente in contatto col computer. Esempi abbastanza chiari sono: dispositivo non collegato, dispositivo spento. Ci sono però altri casi in cui vedrete questo errore, oltre al suddetto problema con l'unità a nastro. Se tentate di scrivere su disco su un file già esistente o provate a leggere un file non esistente sul dischetto collegato, avrete modo di rivedere questo messaggio. Un aspetto importante di questo indicatore è che viene attivato solo quando provate a fare un vero trasferimento di dati. Cosa intendo per

# COMMODORE

vero? Se fate ad esempio una OPEN 1,13, dichiarate solo l'intenzione di fare qualcosa, ma non state trasferendo dei dati e la ST non vi dirà molto. Se invece fate una PRINT# allora, se le condizioni sono quelle descritte, potreste avere un -128 nella ST. Anche se faceste una INPUT#1 o una GET#1 e non avete una unità collegata, stesso messaggio. In questo caso, però, se almeno una unità è collegata (esempio l'unità a nastro), allora il C-64 si bloccherà se fate una INPUT o una GET da una unità non esistente. Il solo modo di uscire da questa condizione è quello di fare una STOP/RESTORE. In conclusione: il C-64 è in grado di dirvi se una unità non è presente per ricevere dati, ma non è in grado di dirvi se una unità è pronta a mandare dei dati. In quest'ultimo caso il C-64, dimostrando la pazienza di Giobbe, aspetterà sine die una risposta dall'unità non presente. Questo non succede però se non ci sono unità attaccate alla porta seriale.

Direi che abbiamo visto praticamente tutti gli indicatori possibili nella variabile di Status e che parlare di questa preziosa variabile è servito anche a chiarirci le idee sul come il C-64 comunica con il mondo esterno. Vedremo nei prossimi numeri di Parlare un po' più dettagliatamente sul come il C-64 parli ed ascolti dalle sue periferiche. Ciao ciao!



## POKE IN MODO 'REVERSE' PER VIC-20

Se state stampando qualcosa sullo schermo, è facile utilizzare il REVERSE mode, premendo il tasto di Ctrl e Reverse On. Quando però state facendo delle POKES nella memoria del video, è più difficile attivare questo meccanismo. Provate a fare POKE 199,1 per andare in Reverse e POKE 199,0 per uscire dal Reverse.

## BEEP per VIC-20

È molto utile, nei programmi, poter avvertire l'utilizzatore di aver accettato il suo INPUT, magari con un leggero suono. Se state usando le istruzioni GET o WAIT per leggere qualcosa da tastiera potete aggiungere le seguenti istruzioni, per produrre un BEEP di ricezione:

```
10 IF PEEK(197)<>64 THEN POKE 36878,15:POKE 36876,232: FOR
D=1 TO 10: NEXT
```

Aggiungendo il RETURN se la usate come subroutine. Se usate INPUT basta togliere l'IF iniziale.

# COMMODORE

## BEEP per C-64

Se volete una subroutine che produca un lieve BEEP quando voi lo volete (per esempio dopo un INPUT), eccovela. Potete modificare l'altezza del suono usando valori diversi da 75 alla linea 1025 mentre la durata del BEEP è regolata dalla linea 1050.

READY.

```
1000 REM ROUTINE DI BEEP PER C-64
1010 :
1020 S=54272: POKES,150
1025 POKES+1,75
1030 POKES+5,0:POKES+6,240:POKES+24,15
1040 POKES+4,17
1050 FORI=1TO200:NEXTI
1060 POKES+24,0:RETURN
```

READY.

## LEGGERE UNA CARTRIDGE SUL VIC-20

Avete mai provato a leggere quello che ci sta dentro una cartridge? Per esempio, se avete un'avventure, i vari messaggi etc? Eccovi un programmino che può aiutarvi. E' molto semplice, e non necessita di nessuna spiegazione, salvo il fatto che se date un "1" durante il listato, lo schermo andrà nero, un "2" lo farà bianco ed un "3" rallenterà per un poco la stampa.

READY.

```
5 REM LETTURA CARTRIDGE PER VIC-20
6 :
10 FORI=17000 TO 40000
20 A=PEEK(I)
30 PRINT CHR$(A);
40 GETA$
50 IF A$="1" THEN POKE 36879,8
60 IF A$="2" THEN POKE 36879,25
70 IF A$="2" THEN FORB=1TO10000:NEXTB
80 NEXT I
```

# COMMODORE



COMMODORE PLAY



## PROBLEMA N.1: Corto e strutturato

Osservate questo listato pubblicato su una rivista americana:

```
10 let X=0
20 print X,2^X
30 let X=X+1
40 if X=18 then 60
50 goto 20
60 end
```

Certamente non presenta grosse difficoltà ne' sul piano dei risultati ne' su quello della Programmazione. Dopo aver capito che cosa deve mostrare sullo schermo, sapreste riscrivere il programma in modo più corto e più strutturato (cioè più leggibile)?

## PROBLEMA N.2: Un buco nell'inventario

Mettetevi nei Panni di un Programmatore di un grande magazzino incaricato di realizzare un programma di inventario. Per ciascun articolo negli scaffali è memorizzato un record sull'archivio di magazzino contenente la quantità ordinata nel mese, quella ricevuta e quella venduta. Per ottenere un tabulato che a fine mese fornisca l'inventario di tutti gli articoli movimentati nel mese (ordinati, ricevuti o venduti, questi ultimi registrati in negativo) avete scritto il seguente programma

```
FOR I=1 TO 1000
IF ORDINATO + RICEVUTO + VENDUTO <> 0
THEN PRINT "Articolo" I, ORDINATO, RICEVUTO, VENDUTO
NEXT I
```

Dopo alcuni mesi il Direttore vi chiama mostrandovi un tabulato dove manca un articolo (i Biscotti del Nonno) di sicuro movimentato nel mese. Sapreste dire cosa è successo e correggere il programma in modo che l'errore non si ripeta?

# COMMODORE

## PROBLEMA N.3: La spia nell'accampamento

Una spia ha il compito di introdursi nell'accampamento nemico: nascosta in un cespuglio ascolta le conversazioni tra la sentinella ed i soldati nemici che tornano alla spicciolata.

"Dodici"-grida la sentinella. "Sei"-risponde un soldato come parola d'ordine e passa avanti. Ne arriva un altro: "Dieci"- la sentinella. "Cinque"- il soldato: avanti. Arriva un terzo: "Otto", "Quattro" e avanti. Quindi la spia è convinta di aver appreso il meccanismo e ne ha una conferma, almeno crede, quando passa un altro soldato che apostrofato dalla sentinella con un imperioso "Sei" risponde immediatamente con "Tre" e passa avanti.

A questo punto, più che sicura, la spia esce allo scoperto e viene apostrofata con un "Quattro" dalla sentinella con il fucile spianato. "Due"-risponde subito la spia, fischiando con naturalezza, ma una pallottola la stende stecchita, mentre nell'accampamento risuona il grido d'allarme. Peccato, la missione è andata male: avrebbe dovuto rispondere "Sette". Perché?

## SOLUZIONE GIOCHI

La chiave del problema non è legata al NUMERO richiesto dalla sentinella, ma alla LUNGHEZZA della parola pronunciata. Così, ad esempio la parola "quattro" ha sette lettere.

L'articolo non viene stampato se la quantità venduta è uguale alla somma di ordinato più ricevuto. Bisogna cambiare il testo come segue:

IF Ordinato = 0 OR Ricevuto = 0 OR Venduto = 0 THEN ....

## PROBLEMA N.2

```
40 END
30 NEXT X
20 PRINT X, 2 X
10 FOR X = 0 TO 12
```

## PROBLEMA N.1

# CO.STA MARKETING

A tutti gli amici lettori di SOFTWARE DOSSIER un'offerta a prezzi molto vantaggiosi di prodotti di consumo per personal e home computer.  
Per effettuare l'ordinazione è sufficiente compilare e spedire in busta chiusa a:

**CO.STA SRL - VIA MAZZINI 128 - 56100 PISA - SERVIZIO MARKETING**

il sottostante tagliando, allegando il pagamento dell'importo complessivo in contanti o tramite assegno bancario o vaglia postale. I prezzi sono comprensivi di IVA e oneri di spedizione. Per spedizioni in contrassegno supplemento fisso di lire 3.000.

## DISCHETTI

☐ **Cod. V001**

Minifloppy 5" 1/4 semplice faccia, semplice densità.  
Scatola da 10 dischetti **Lire 45.000**  
Ogni scatola oltre la 1ª sconto 5%.

☐ **Cod. V002**

Minifloppy 5" 1/4 doppia faccia, doppia densità.  
Scatola da 10 dischetti **Lire 55.000**  
Ogni scatola oltre la 1ª sconto 5%.

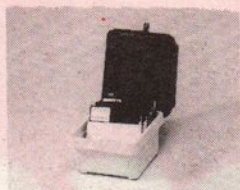
☐ **Cod. 8002**

Floppy 8" doppia faccia, doppia densità. Scatola da 10 dischetti **Lire 65.000**  
Ogni scatola oltre la 1ª sconto 5%.

## CONTENITORI

☐ **Cod. BF01**

Contentore a scatola FLEX BOX per 10 minifloppy con coperchio Rosso, Giallo o Nero **Lire 5.000**



**Cod. C040**

Contentore in plastica da 40 minifloppy, con serratura **Lire 28.000**



**Cod. C090**

Contentore in plastica da 90 minifloppy, con serratura **Lire 35.000**

☐ **Cod. TD01**

Supporto per stampante e contenitore di moduli continui TOP DESK in plastica trasparente **Lire 90.000**

## FLOPPY DISKS

☐ **Cod. C001**

Floppy Disks Album **Lire 8.500**

☐ **Cod. E001**

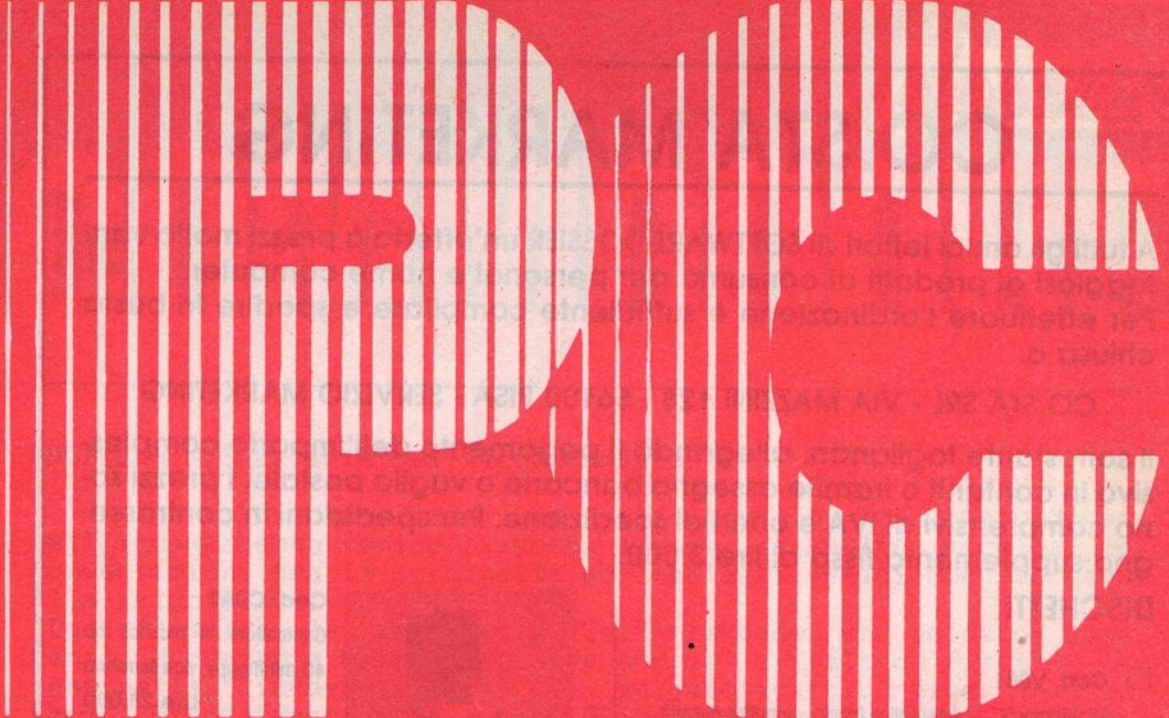
Buste per 2 Floppy Disks 5" 1/4 con etichette in clear antimagnetico **Lire 800**

☐ **Cod. E002**

Buste per Floppy Disks 8" con etichetta in clear antimagnetico **Lire 800**

**Ordine minimo lire 50.000**

| CODICE    | QUANTITÀ | PREZZO TOTALE |                    |
|-----------|----------|---------------|--------------------|
|           |          |               | Nome _____         |
|           |          |               | Cognome _____      |
|           |          |               | Via _____ N° _____ |
|           |          |               | Città _____        |
|           |          |               | _____ CAP _____    |
|           |          |               | Firma _____        |
| TOTALE L. |          |               |                    |



# PERSONAL COMPUTER **CLUB**

**GIOCHI-CONCORSO**

**PROGRAMMI E TRUCCHI**

**PROVE HARDWARE E SOFTWARE**

**PROGETTI SOFTWARE**

**BIBLIOTECA PERSONALE**

**e le rubriche**

**TUTTOCOMMODORE**

**TUTTOSINCLAIR**

**TUTTOMSX**

**OGNI MESE  
IN EDICOLA  
LIRE 3.000**